Opérateur INTE_MAIL_2D

1 But

Définir une courbe dans un maillage 2D. Aux points d'intersection de la courbe ainsi définie avec le maillage pourront être effectués, à l'aide de l'opérateur POST_RELEVE_T, des relevés de valeurs, des calculs (moyennes, invariants, ...) et le stockage du résultat de ces opérations dans un concept de type table.

Le concept produit est de type courbe.

Remarque :
Cette commande sera prochainement supprimée.
Il est conseillé de la remplacer par la commande MACR_LIGN_COUPE

2 Syntaxe

```
crb [courbe] = INTE_MAIL_2D

(   ♦  MAILLAGE = ma,                                         [maillage]

   ◊  /  TOUT      =  'OUI',
      /  GROUP_MA  =  lgrma,                                  [l_gr_maille]
      /  MAILLE    =  lmail,                                  [l_maille]

    ♦ /   I  DEFI_SEGMENT=_F(♦  /  ORIGINE      =(xa,ya),     [l_R]
                                /  NOEUD_ORIG   = noeud,      [noeud]
                                /  GROUP_NO_ORIG= grno,       [group_no]
                             ♦  /  EXTREMITE    =(xb,yb),     [l_R]
                                /  NOEUD_EXTR   = noeud,      [noeud]
                                /  GROUP_NO_EXTR= grno,       [group_no]
                        ),

          I  DEFI_ARC = _F(  ♦  /  CENTRE         =(xc,yc),   [l_R]
                                /  NOEUD_CENTRE   = noeud,    [noeud]
                                /  GROUP_NO_CENTRE= grno,     [group_no]
                             ♦  /  ♦   RAYON   =   r,            [R]
                                   ♦  SECTEUR = (□inf,□sup),  [l_R]
                                /  ♦  /  ORIGINE =(xa,ya),    [l_R]
                                      /  NOEUD_ORIG = noeud,  [noeud]
                                      /  GROUP_NO_ORIG=grno,  [group_no]
                                   ♦  /  EXTREMITE= (xb,yb),  [l_R]
                                      /  NOEUD_EXTR= noeud,   [noeud]
                                      /  GROUP_NO_EXTR=grno,  [group_no]
                              ◊ PRECISION =  /   epsilon ,          [R]
                                             /  0.001,        [DEFAUT]
                              ◊ CRITERE =    /  'RELATIF',    [DEFAUT]
                                             /  'ABSOLU' ,
                        ),

      /  ♦  DEFI_CHEMIN=_F(  ♦  /  MAILLE   = mail,           [l_maille]
                                /  GROUP_MA = grma,           [l_gr_maille]
                         ),

          ◊ /  NOEUD_ORIG    =  noeud,                        [noeud]
            /  GROUP_NO_ORIG =  grno,                         [group_no]

      ◊  PRECISION =     /  epsilon,                                [R]
                         /  0.001,                            [DEFAUT]

      ◊  INFO =          /  1,                                [DEFAUT]
                         /  2,

)
```

3 Opérandes

3.1 Opérande MAILLAGE

♦ MAILLAGE = ma

ma : nom du concept de type maillage sur lequel est repérée la courbe.

3.2 Opérandes TOUT / GROUP_MA / MAILLE

Ne s'appliquent que pour DEFI_SEGMENT et DEFI_ARC.

◊ / TOUT = 'OUI',

Le repérage s'effectue sur tout le maillage.

/ GROUP_MA = lgrma,

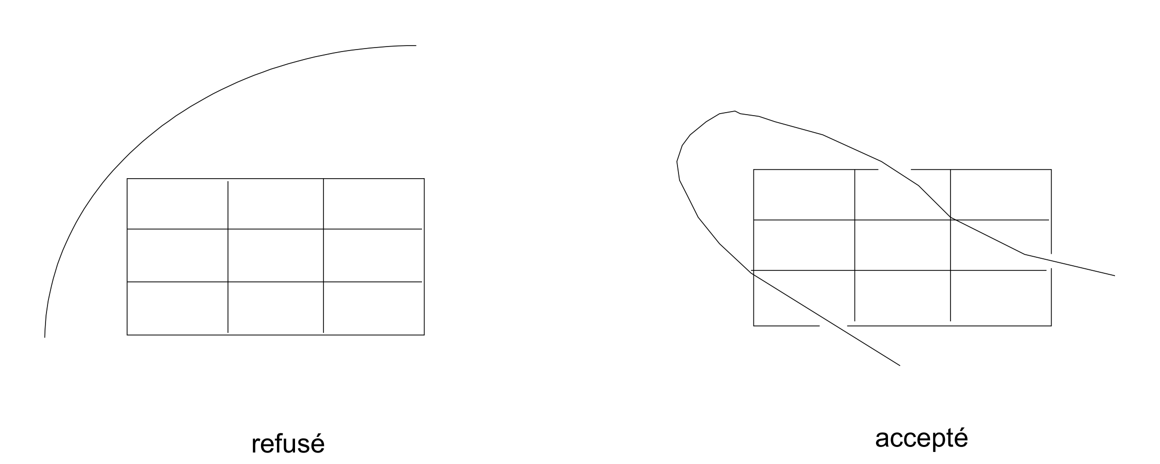
Le repérage s'effectue seulement sur les groupes de mailles de la liste lgrma.

/ MAILLE = lmail,

Le repérage s'effectue seulement sur les mailles de la liste lmail.

3.3 Chemin : Mots clé DEFI_SEGMENT / DEFI_ARC / DEFI_CHEMIN

Le chemin peut être défini par une ou plusieurs occurrences de DEFI_CHEMIN, ou bien une ou plusieurs occurrences de DEFI_ARC et DEFI_SEGMENT ensembles.
Si une occurrence de DEFI_SEGMENT, DEFI_ARC ou DEFI_CHEMIN définit une courbe qui ne coupe pas le maillage, un message d'erreur fatale est émis.



refusé accepté

3.3.1 Mot clé DEFI_SEGMENT

Mot clé facteur dont chaque occurrence définit un segment de droite par la donnée de ses points origine et extremite (sous forme de coordonnées ou de noms de nœuds ou group_no). Le segment est orienté de l'origine vers l'extrémité.

3.3.2 Mot clé DEFI_ARC

Mot clé facteur dont chaque occurrence définit un arc de cercle par la donnée de son centre et, soit d'un rayon et d'un secteur angulaire, soit de 2 points origine et extrémité de l'arc.

Le centre du cercle est introduit par l'un des 3 mots clés :

CENTRE	=(xc, yc),	coordonnées du centre du cercle,
NOEUD_CENTRE	= noeud,	nom du nœud centre du cercle,
GROUP_NO_CENTRE	= grno,	nom du group_no contenant le seul nœud centre,

Le rayon du cercle est introduit par le mot clé :

RAYON = r   avec r > 0

Le secteur angulaire par le mot clé :

SECTEUR = $(\theta_{\text{inf}}, \theta_{\text{sup}})$ avec $(\theta_{\text{inf}}, \theta_{\text{sup}})$ angles en degrés vérifiant $-180 < \theta_{\text{inf}} \leq \theta_{\text{sup}} \leq 180$

Le point origine de l'arc par l'un des 3 mots clés :

ORIGINE	=(xa, ya),	coordonnées du nœud origine,
NOEUD_ORIG	= noeud,	nom du nœud origine,
GROUP_NO_ORIG	= grn o,	nom du group_no contenant le seul nœud origine.

Le point extrémité de l'arc par l'un des 3 mots clés :

EXTREMITE	=(xb, yb),	coordonnées du nœud extrémité,
NOEUD_EXTR	= noeud,	nom du nœud extrémité,
GROUP_NO_EXTR	= grno,	nom du group_no contenant le seul nœud extrémité.

PRECISION = epsilon

Précision valable pour une occurrence du mot clé facteur DEFI_ARC. Permet de surcharger la précision valable pour toute la commande [§3.3].

```
CRITERE =     /  'RELATIF',                    [DEFAUT]
              /  'ABSOLU' ,
```

Si q est la quantité recherchée avec une précision ε , alors l'intervalle de recherche est :

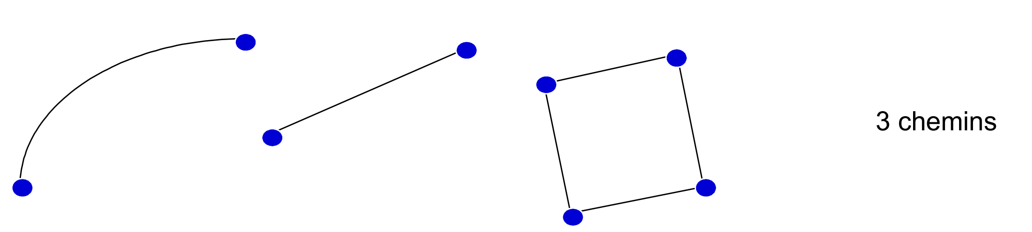
$[q(1-\varepsilon), q(1+\varepsilon)]$ en 'RELATIF'
$[q-\varepsilon, q+\varepsilon]$ en 'ABSOLU'

3.3.3 Mot clé DEFI_CHEMIN

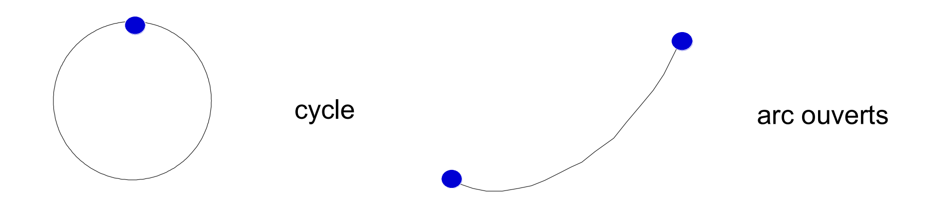
Mot clé facteur dont chaque occurrence admet pour argument une liste de noms de mailles (mot clé MAILLE) ou une liste de noms de groupes de mailles (mot clé GROUP_MA). Ces mailles étant du type SEG2 ou SEG3.

Le chemin (ou éventuellement les chemins) est constitué à partir de la réunion des différentes mailles. INTE_MAIL_2D analyse cet ensemble pour déterminer la topologie. Il détecte :

- l'existence ou non de plusieurs chemins indépendants :



- pour chaque chemin, on distingue les cycles et les arcs ouverts :



Les chemins sont orientés à partir de la maille de plus petit numéro pour les cycles et à partir de la maille extrémité de plus petit numéro pour les arcs ouverts. Ce numéro correspond à l'ordre d'apparition dans le fichier de maillage.

L'utilisateur peut néanmoins imposer le nœud origine du chemin par le mot clé `NOEUD_ORIG` (nom du nœud origine) ou `GROUP_NO_ORIG` (nom du `group_no` formé du seul nœud origine).

3.4 Opérande PRECISION

◊ `PRECISION = epsilon`

Mot clé facultatif permettant à l'utilisateur de définir le seuil en dessous duquel 2 points sont considérés comme confondus.

3.5 Opérande INFO

Permet d'obtenir des impressions sur le fichier `'MESSAGE'`.

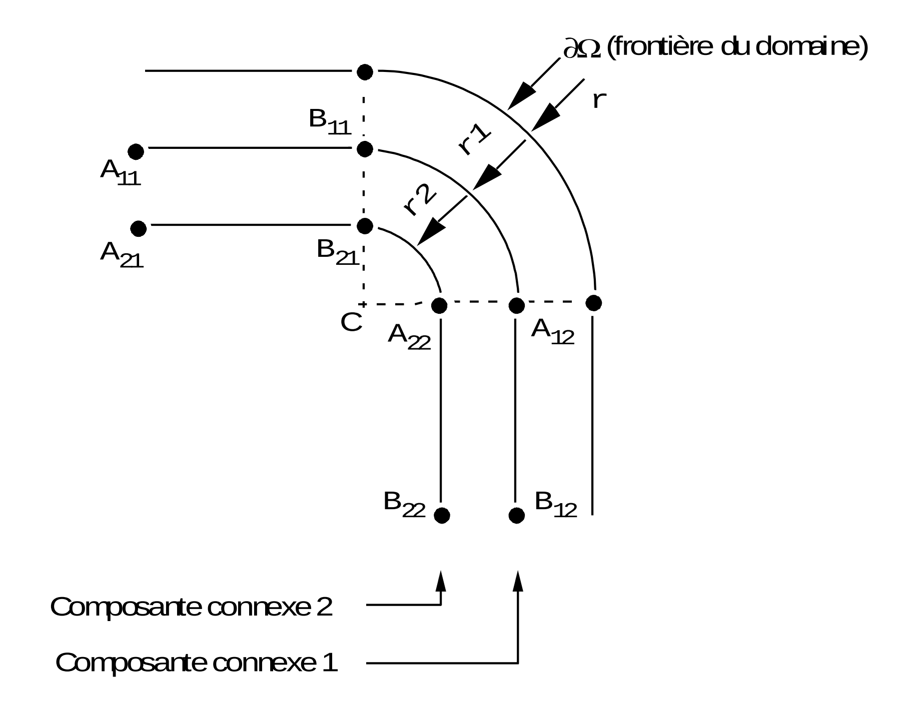
`INFO = 1,` pas d'impressions
`INFO = 2,` impression des mailles traversées par le chemin

4 Possibilités et limites d'utilisation

4.1 Courbes obtenues comme réunion de segments de droite et/ou d'arcs de cercle

Exemple 1

Supposons que la frontière du domaine se réduise localement à 2 segments de droite et un arc de cercle et que l'utilisateur s'intéresse au comportement de la structure au voisinage de cette frontière. Il pourra alors définir des courbes telles que A_{11}, B_{11}, A_{12}, B_{12} ou A_{21}, B_{21}, A_{22}, B_{22}.



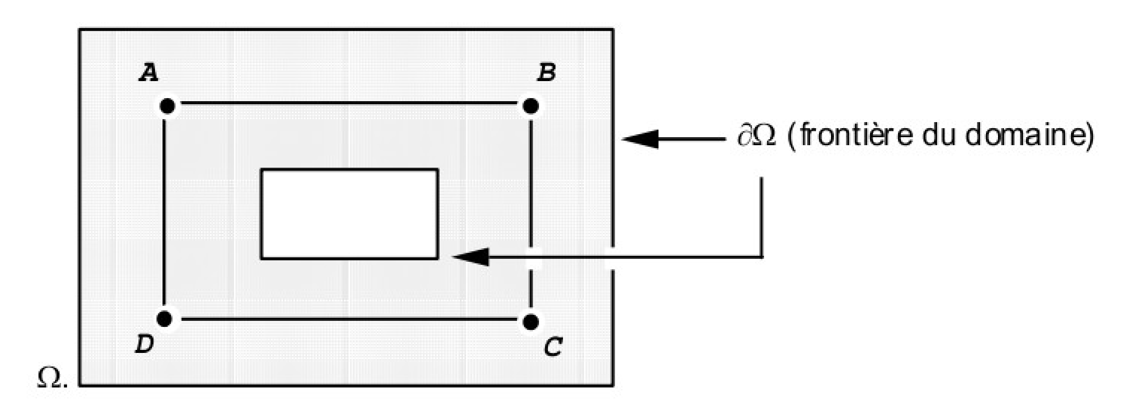

```
Coin = INTE_MAIL_2D

(  MAILLAGE = nom du maillage,

   DEFI_SEGMENT = (
   _F  (  ORIGINE = (xA11, yA11),EXTREMITE = (xB11, yB11),),
   _F  (  ORIGINE = (xA21, yA21),EXTREMITE = (xB21, yB21),),
   _F  (  ORIGINE = (xA12, yA12),EXTREMITE = (xB12, yB12),),
   _F  (  ORIGINE = (xA22, yA22),EXTREMITE = (xB22, yB22),),),

    DEFI_ARC = (
   _F  (  CENTRE =(xc1 , yc1),
          RAYON  = r1 , SECTEUR = (  0., 90.),  ),
   _F  (  CENTRE =(xc2 , yc2),
          RAYON  = r2 , SECTEUR = (  0., 90.),  ),),
)
```

Exemple 2 : Etude au voisinage d'un trou intérieur dans un domaine

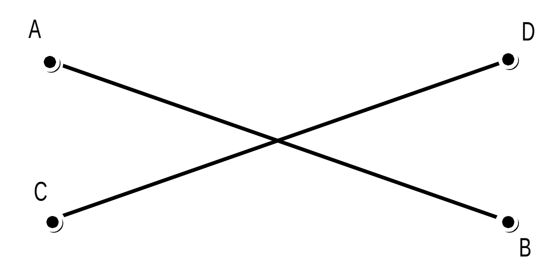


La courbe ABCD est définie comme réunion de 4 segments de droite.

```
tour = INTE_MAIL_2D
    (  MAILLAGE = carré_percé,
       DEFI_SEGMENT = (
       _F (   ORIGINE = (xA, yA), EXTREMITE = (xB, yB)    ),
       _F (   ORIGINE = (xD, yD), EXTREMITE = (xC, yC)    ),
       _F (   ORIGINE = (xA, yA), EXTREMITE = (xD, yD)    ),
       _F (   ORIGINE = (xB, yB), EXTREMITE = (xC, yC)    ),),
    )
```

Exemple 3 : une limite d'utilisation

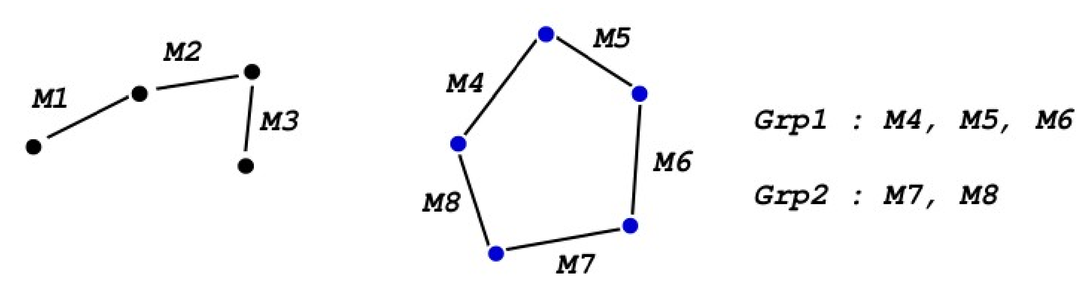
Elle concerne la possibilité d'intersection des segments et/ou des arcs en des points qui ne sont pas une de leur extrémité.



Ce cas ne fait l'objet d'aucun traitement particulier. Les 2 segments de droite sont pris comme deux morceaux parfaitement indépendants d'un même concept. Leur intersection est ignorée. La gestion de tels cas est à la charge de l'utilisateur.

4.2 Courbes obtenues comme réunion de mailles 1D du domaine

Exemple 1

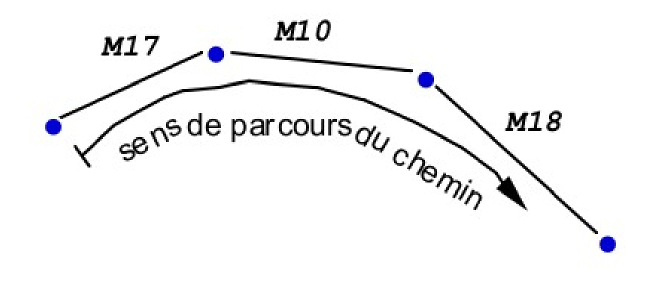


La courbe se réduit au chemin simple constitué des mailles `M1`, `M2`, `M3` et au cycle correspondant aux groupes de mailles `Grp1` et `Grp2`. Le chemin simple et le cycle constituent les deux composantes connexes de la courbe.

L'ordre dans lequel sont parcourues les mailles d'un chemin dépend de la numérotation des mailles.

Dans le cas d'un chemin simple (arc ouvert) c'est la numérotation des mailles "extrémité" qui détermine l'ordre de parcours.

Par exemple :

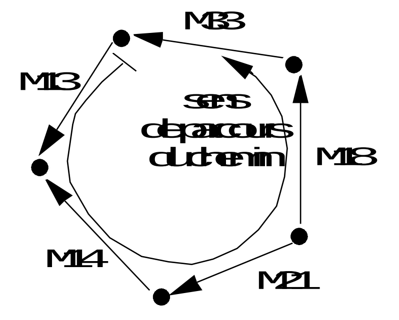



```
Parcours = INTE_MAIL_2D

    (   MAILLAGE = Maya,

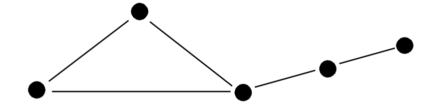
        DEFI_CHEMIN = _F (   MAILLE        = ('M1','M2','M3'),
                                   GROUP_MA = ('Grp1', 'Grp2'), )
    )
```

Dans le cas d'un cycle, le chemin est parcouru en partant de la maille de plus petit numéro et dans le sens de la connectivité de cette maille



Exemple 2 : une restriction importante
L'opérateur suppose que les cycles et les chemins simples sont disjoints, ainsi le cas suivant est illicite.

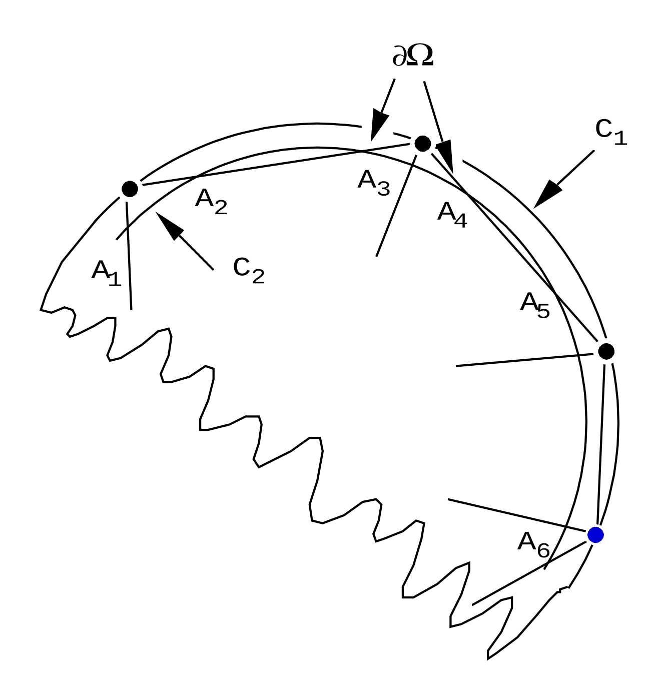
Ce cas n'est pas détecté mais doit être évité.



Exemple 3 : une limite d'utilisation pour le bord courbe

Pour un domaine dont une partie de la frontière est un arc de cercle, il est tentant de définir un arc de cercle, au moyen du mot clé `DEFI_ARC`, qui couperait le maillage au voisinage de la frontière. Malheureusement, le bord du maillage n'est pas un arc de cercle mais un polygone et le chemin défini peut sortir du maillage.

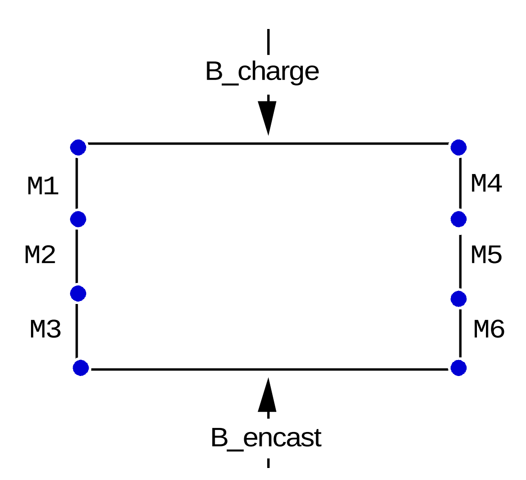
Par exemple :



C_1 : cercle extérieur
C_2 : cercle intérieur
$d\,\Omega$: ligne brisée

- L'arc de cercle C_1, qui coïncide avec une partie de □□, ne coupe le maillage de □ qu'en des nœuds : `INTE_MAIL_2D` donnera une intersection vide.
- L'arc de cercle C_2 est strictement inclus dans □ mais dans le maillage de □ `INTE_MAIL_2D` repérera 2 trous.

Exemple 4 : étude sur la frontière d'une plaque




```
Bord = INTE_MAIL_2D

    (   MAILLAGE = plaque,

        DEFI_CHEMIN = _F (   MAILLE = ('M1','M2','M3','M4','M5','M6',),
                              GROUP_MA = ('B_charge',   'B_encast'),)
    )
```

Pour étudier le comportement de la structure sur son bord encastré, il est conseillé de définir une autre courbe comme suit :

```
Bord_Enc = INTE_MAIL_2D

    (   MAILLAGE = plaque,

        DEFI_CHEMIN = _F (GROUP_MA = ('B_encast' ),),
    )
```